Image processing with learning capabilities for color images

Gulzhan Zhassulanbaikyzy

Image processing with learning capabilities for color images

Gulzhan Zhassulanbaikyzy
Université Grenoble Alpes, Laboratoire Jean Kuntzmann

July 2017

Abstract

The application of anisotropic diffusion equation for color image processing is discussed in this master thesis. Nonlinear PDEs of diffusion type are a common tool for color image processing and image filtering. These PDEs are designed in such a way that diffusion eliminates the noise but preserves the significant edges and features of the image. A diffusion tensor with learning abilities was designed in to provide these features. This model was previously applied to grey-scale images and gave excellent results [9]. In this work this model is extended to color images. The proposed algorithm for color image denoising is implemented in the form of C++ scripts. In order to evaluate our proposed algorithm, comparisons with Chambolles's projection algorithm [21, 6] and the more recent DA3D algorithm [38] are performed. The source code of both algorithms the description of the methods and the test images were borrowed from the IPOL online resources. The results confirm the good performance of the diffusion tensor approach for color images.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Image denoising has been one of the most important and widely studied problems in image processing and computer vision. The need to have a very good image quality is increasingly required with the advent of the new technologies in various areas such as multimedia, medical image analysis, aerospace, video systems and others. Indeed, the acquired image is often marred by noise which may have a multiple origins such as: thermal fluctuations; quantify effects and properties of communication channels. It affects the perceptual quality of the image, decreasing not only the appreciation of the image, but also the performance of the task for which the image has been intended. The challenge is to design methods, which can selectively smooth a degraded image without altering edges, losing significant features and producing reliable results. Traditionally, linear models have been commonly used to reduce noise. It is shown that these methods perform well in the flat regions of images, but do not preserve edges and discontinuities which are often smeared out. In contrast, nonlinear models can handle edges in a much better way than those linear models. Many approaches have been proposed to remove the noise effectively while preserving the original image details and features as much as possible. In the past few years, the use of non linear PDEs methods involving nonlinear diffusion has significantly grown and becomes an important tool in contemporary image processing. The key idea behind the nonlinear diffusion is to incorporate an adaptive smoothness constraint in the denoising process.

Nonlinear diffusion equations for image processing were first proposed by Perona and Malik [37]. It was later improved by Catté, Lions, Morel and Coll [15] and refined by Alvarez, Lions and Morel [28]. The derivation and

application of diffusion tensors instead of a scalar diffusion coefficient were proposed by Cottet and Germain [10] and Weickert [24]. The work in [10] was further improved by Cottet and El-Ayyadi in [9]. In the later model, time-delay regularization instead of spatial regularization is used to construct the diffusion tensor. This gives a natural adaptivity of the diffusion and allows to obtain processed images on steady-states of the models. The selected steady states are based on a contrast parameter involved in the time-delay regularization. The model also bears some analogy with neural networks which obey hebbian rules to update synaptic weights [8]. In the present work we will demonstrate how an anisotropic diffusion equation, similar to that studied in [9], can be used to perform filtering of color images. The application of this approach to color images is done by splitting it into the three color channels Red, Green and Blue and filtering each channel separately.

In order to evaluate the proposed algorithm in comparison with other color image denoising algorithms, the database of IPOl (Image Processing On Line) is used. IPOL is an open access journal which publishes state-of-the-art image processing and image analysis algorithms emphasizing the role of mathematics as a source for algorithm design. It provides the detailed description of the published algorithms together with their software implementation in the form of C, C++ or Matlab codes. We will make extensive comparisons with two papers published in the journal, Chambolle's projection algorithm [21] and the Data Adaptive Dual Domain Denoising (DA3D) algorithm [38]. The tests will be conducted on noisy color images obtained from noise-free original images by adding Gaussian noise of several standard deviation δ. Two image quality metrics, the PSNR (peak signal to noise ratio) and RMSE (root mean square error), will be used to evaluate the quality of the restored image. All experiments and obtained results are described in Chapter 4.

This master thesis is organized as follows. In Chapter 2, we give an overview of diffusion models for image processing, with more details on time-delay anisotropic diffusion models. In Chapter 3, the application and implementation details of the proposed method for color images are presented. Numerical experiments and results on real photographic images are shown in Chapter 4. Conclusion and bibliography are given in Chapters 5 and 6 respectively.

Chapter 2

Image processing techniques based on partial differential equations

Image denoising has been one of the most important and widely studied problems in image processing and computer vision. Indeed, the acquired image is often marred by noise which may have a multiple origins such as: thermal fluctuations; quantify effects and properties of communication channels. It affects the perceptual quality of the image, decreasing not only the appreciation of the image, but also the performance of the task for which the image has been intended. The challenge is to design methods, which can selectively smooth a degraded image without altering edges, losing significant features and producing reliable results. In this context PDE based techniques for image processing have played central role as PDE-based methods are one of the mathematically best-founded techniques in image processing and provides deep mathematical results with respect to well-posedness. Furthermore, it provides stable algorithm for image processing.

PDE-based image processing techniques are mainly used for smoothing and restoration purposes. In image processing we need to cover two important requirements, firstly, it is necessary to smooth homogeneous regions of the image and at the same time we should preserve the image edges and features carefully. In order to meet two significant requirements, Marr and Hildreth [31], later formalized by Witkin [42], Koenderink [27] and Canny [5], uses a low-pass filtering obtained by convoying the image with Gaussian of increasing variance. Koenderink [27] soon found out that

the convolution of the image with a Gaussian at each scale is the solution of the heat equation itself with the image as initial state.

Let us consider an image $u_0(x)$, where $x = (x_1, x_2)$ denotes space coordinates, the scale-space analysis associated with u_0 consists in solving the next system

$$\partial_t u - \bigtriangledown^2 u = 0 \quad u(x, o) = u_0(x) \tag{2.1}$$

It has a unique solution

$$u(x,t) = \begin{cases} u_0(x) & t = 0 \\ (G_{\sqrt{2t}} * u_0)(x) & t > 0 \end{cases} \tag{2.2}$$

provided that the function satisfies $|u(x,t)| \leq M \exp(a|x|^2), \quad M > 0$, it depends continuously on the initial condition u_0 with respect to $||\cdot||_{L^\infty(R^2)}$, and it meets the maximum-minimum principle $\inf_{R^2} \leq u(x,t) \leq sup_{R^2} u_0$ on $R^2 \times [0, \infty)$. The point x is an edge for the scale t where $| \bigtriangledown u(x,t)|$ is large and $\bigtriangledown^2 u(x,t)$ changes sign.

Gaussian smoothing is very attractive for image denoising in terms of its simplicity and effectiveness, but at the same time it has some disadvantages as Gaussian filter does not smooth only noise but also smooths everything along with it and does not preserve edges and artifacts of the image [42, 43]. Most of the shortcomings of linear diffusion processes can be avoided through nonlinear diffusion models.

2.1 Nonlinear (anisotropic) diffusion models

In order to avoid the problems like blurring and localization which appears in the case of linear diffusion filtering, Perona and Malik [37] in 1987 propose a nonlinear diffusion method which they called "anisotropic". The model accomplishes this by applying a process that reduces the diffusivity in places having higher likelihood of being edges. This likelihood is measured by a function of the local gradient $| \bigtriangledown u|$. Another words they propose to replace the heat equation by a nonlinear equation. The model can be written as

$$\partial_t u - div \cdot (g(| \bigtriangledown u|) \bigtriangledown u) = 0, \quad \partial_n u = 0, \quad u(x,0) = u_0(x) \tag{2.3}$$

where $\partial_n u = 0$ denotes homorgeneous Neumann boundary conditions. In this model the diffusuvity has to be such that $g(| \bigtriangledown u|^2) \to 0$ when $| \bigtriangledown u| \to \infty$ and $g(| \bigtriangledown u|^2) \to 1$ when $| \bigtriangledown u| \to 0$. The idea is that the smoothing

process obtained by the equation is conditional: if $g| \bigtriangledown u|$ is big, then the diffusion will be low and therefore the exact loclization of the edges will be preserved. If $g| \bigtriangledown u|$ is small then the diffusion will tend to smooth still more around (x_1, x_2). One of the diffusivities Perona and Malik proposed is

$$g(| \bigtriangledown u|^2) = \frac{1}{1 + | \bigtriangledown u|^2/\lambda^2}, \quad \lambda > 0 \tag{2.4}$$

where λ is a threshold (contrast) parameter that separates forward (low contrast) and backward (high contrast) diffusion [41]. Although Perona and Malik name their filter anisotropic, it should be noted that – in our terminology – it would be regarded as an isotropic model, since it utilizes a scalar-valued diffusivity and not a diffusion tensor.

The experiments of Perona and Malik were visually very impressive: edges remained stable over a very long time. It was demonstrated [37] that edge detection based on this process clearly outperforms the linear Canny edge detector,even without applying non-maxima suppression and hysteresis thresholding. This is due to the fact that diffusion and edge detection interact in one single process instead of being treated as two independent processes which are to be applied subsequently.

Despite of the practical success of the Perona-Malik model, it involves some serious theoretical problems: (a) None of the classical well-posedness frameworks is applicable to the Perona-Malik model, i.e. we can not ensure well-posedness results [33, 25]; (b) Uniqueness and stability with respect to the initial image should not be expected, i.e. solvability is a difficult problem, in general [18, 19, 26, 35, 15]; (c) The regularizing effect of the discretization plays too much of an important role in the solution [16, 3]. The latter is perhaps the key element in the success or failure of the model. Most practical applications work very well provided that the numerical schemes stabilize the process through some implicit regularization.

The model which has been proposed by Catté, Lions, Morel and Coll [15] is a synthesis of Malik and Perona's ideas which avoids the abovementioned difficulties. These authors replace the diffusivity $g(| \bigtriangledown |^2)$ of the Perona and Malik model by $g(| \bigtriangledown u_\sigma|^2)$ with $u_\sigma := G_\sigma * u$ and establish existence, uniqueness and regularity of a solution for $\sigma > 0$. Their proposed model is therefore

$$\partial_t u - div(g(| \bigtriangledown u_\sigma|^2) \bigtriangledown u) = 0 \quad \partial_n u = 0, \quad u(x, 0) = u_0(x) \tag{2.5}$$

However, this last model keeps some of the drawbacks of the previous model as it has no clear geometric interpretation, because the term inside

the divergence is hybrid and combines the estimate of the gradient. In practice, after the Gaussian filtering, the term $g(|\bigtriangledown u_\sigma|^2)$ allows detection of the location of the main edges and prevents excessive diffusion at these location. But the small noise or fluctuation will be smooth enough and can be diffused away.

It was refined by Alvarez, Lions and Morel further. They proposed and studied a class of nonlinear parabolic differential equations of the form:

$$\partial_t u - g(|G * \bigtriangledown u|)|\bigtriangledown u| div(\frac{\bigtriangledown u}{|\bigtriangledown u|}) = 0 \quad \partial_n u = 0 \quad u(x,0) = u_0(x) \quad (2.6)$$

The degenerate diffusion term $|\bigtriangledown u| div(\frac{\bigtriangledown u}{|\bigtriangledown u|})$ diffuses u in the direction orthogonal to its gradient $\bigtriangledown u$ and prevents diffusion in the direction of $\bigtriangledown u$. The term $g(|G * \bigtriangledown u|)$ is used for edge enhancement and it controls the speed of the diffusion.

2.2 Nonlinear diffusion filtering with a Diffusion tensor

All nonlinear diffusion filters that we mentioned before utilize a scalar-valued diffusivity g which is adapted to the underlying image structure. Therefore, they are isotropic and the flux $j = -g \bigtriangledown u$ is always parallel to $\bigtriangledown u$. But it is desirable to direct the flux towards the orientation of interesting features. It can not be done by a scalar diffusivity anymore, so it may be accomplished by using a diffusion tensor instead of a scalar diffusivity [10, 41, 8].

All Anisotropic diffusion filters usually apply spatial regularization strategies, only exception is the time-delay regularization of Cottet and El-Ayyadi [9] which is discussed deeply in the next Section 2.3.

There are two main representatives of anisotropic diffusion processes. The first one offers advantages at noisy edges, whereas the second one is well-adapted to the processing of one-dimensional features. They are called edge-enhancing anisotropic diffusion and coherence-enhancing anisotropic diffusion, respectively.

2.2.1 Edge-enchancing anisotropic diffusion

In the interior of a segment the nonlinear isotropic diffusion equation (2.5) behaves almost like the linear diffusion filter (2.1), but at edges diffusion

is inhibited. Therefore, noise at edges cannot be eliminated successfully by this process. To overcome this problem, a desirable method should prefer diffusion along edges to diffusion perpendicular to them.

Anisotropic models do not only take into account the modulus of the edge detector $\bigtriangledown u_\sigma$, but also its direction. In this purpose, there is constructed the orthonormal system of eigenvectors v_1, v_2 of the diffusion tensor D such that they reflect the estimated edge structure:

$$v_1 \parallel \bigtriangledown u_\sigma, \quad v_2 \perp \bigtriangledown u_\sigma \tag{2.7}$$

To smooth across the edge instead of along the edge, Weickert propose to choose the corresponding eigenvalues λ_1 and λ_2 as follows:

$$\begin{cases} \lambda_1(\bigtriangledown u_\sigma) := g(|\bigtriangledown u_\sigma|^2) \\ \lambda_2(\bigtriangledown u_\sigma) = 1 \end{cases} \tag{2.8}$$

Commonly, $\bigtriangledown u$ does not match with one of the eigenvectors of the diffusion tensor D as long as $\sigma > 0$. Hence, this model behaves really anisotropic. If we let the regularization parameter σ tend to 0, we finally come to the isotropic Perona–Malik process.

2.2.2 Coherence-enhancing anisotropic diffusion

Another motivation for introducing anisotropy into diffusion processes arises from the wish to process one-dimensional features such as line-like structures. It was presented by Cottet and Germain [10], they constructed a diffusion tensor with eigenvectors as in (2.7) and corresponding eigenvalues

$$\begin{cases} \lambda_1(\bigtriangledown u_\sigma) := 0 \\ \lambda_1(\bigtriangledown u_\sigma) := \frac{\eta|\bigtriangledown u_\sigma|^2}{1+(\bigtriangledown u_\sigma|/\sigma)^2}, \quad \eta > 0 \end{cases} \tag{2.9}$$

This is a process diffusing solely perpendicular to $\bigtriangledown u_\sigma$. For $\sigma \to 0$, we observe that $\bigtriangledown u$ becomes an eigenvector of diffusion tensor D with corresponding eigenvalue 0. Therefore, the process stops completely. In this sense, it is not intended as an anisotropic regularization of the Perona–Malik equation. When u has support around a one-dimentional curve, provided its curvature does not vary too much on a scale of the order of σ. Therefore, it can be expected that the diffusion will not affect smooth one-dimentional objects, and σ appears again as a scale parameter which will determine the minimal size of the details that one wishes to keep in the image. On the other hand, if the signal is noisy, then the direction of $\bigtriangledown u_\sigma$ and $\bigtriangledown u$ will not show any coherence, and $\triangle_\sigma$ will act as an isotropic diffusion.

2.3 Time-delay anosotropic diffusion

In contrast with the above anisotropic diffusion filters which use spatial regularization strategies, Cottet and El-Ayyadi [9] proposed a the time-delay regularization. Their model is the combination of two tools: anisotropic nonlinear diffusion and time-delay regularization. It is known from the previous observations that anisotropic diffusion tensors allow a better tracking of the edges in comparison with a scalar diffusion models. Furthermore, time-delay regularization is an alternative to spatial regularization in the construction of diffusion coefficients from the image itself. The model is driven by a contrast parameter which selects steady states. These steady states consist in images made of homogeneous patterns (with constant grey levels) separated by fronts with stiffness controlled by the contrast parameter. The model has the capability of restoring these steady states as asymptotic states, thus avoiding the, sometimes delicate, choice of a stopping time. In practical implementation the stopping time can be obtained from the value of the residual between two successive iterations on the model.

2.3.1 Description of the model

Let us consider an initial image in the unit square $\Omega =]-1,+1]^2$ whose grey level is given by a function u_0 with values in $[-1,+1]$. The filter is based on the next system of differential equations:

$$\frac{\partial u}{\partial t} - div(L \bigtriangledown u) = 0 \tag{2.10}$$

$$\frac{\partial L}{\partial t} + L = F(\bigtriangledown_\sigma u) \tag{2.11}$$

where
$\bigtriangledown_\sigma u = \bigtriangledown(u * f_\sigma), \quad f_\sigma(y) = \sigma^{-2} f(\frac{y}{\sigma}), \quad \int f dx = 1.$
The system is supplemented with initial values u_0 (noisy image) and diffusion matrixL_0 and periodic boundary conditions. But boundary conditions are no significant conditions in image processing they are only dictated by our wish to get rid of any technical unessential difficulties.

In the equation (2.10) u is a evolving image, L is a 2×2 diffusion matrix responsible for anisotropy, and F is a 2×2 anisotropy force matrix which is a function on an image gradient. It is an anisotropic diffusion equation and the diffusion matrix takes into account information, as time goes on, from the gradient of u.

σ is a positive smoothing parameter needed for the mathematical well-posedness of the system, in precise $\sigma = 0$ (for which $\bigtriangledown_\sigma u = \bigtriangledown u$) proved to be satisfactory. It is very important to mention that the parameter should be considered as independent case, in contrast with other diffusion models where the smoothing parameter has to be adjusted in particular to the noise level of the image.

Let us now indicate the smoothness assumption on f and F:
F is a nonnegative symmetric matrix:

$$\langle F(v)\omega, \omega\rangle \leq 0, \quad \forall v, \omega \in R^2 \tag{2.12}$$

F and its derivatives are bounded:

$$|F(v)| + |\bigtriangledown F(v)| \leq C, \quad \forall v \in R^2 \tag{2.13}$$

$$f \quad and \quad its \quad derivatives \quad are \quad bounded. \tag{2.14}$$

The goal of the (2.13) is clearly to avoid anti-diffusion in (2.10): if one starts from a positive diffusion matrix

$$L_0 \geq \alpha Id, \quad \alpha > 0 \tag{2.15}$$

the explicit integration of (2.11) combined with (2.13) yields

$$L(:, t) \geq \alpha e^{-t} Id \tag{2.16}$$

and (2.10) is a parabolic equation. For details of the proof of the well-posedness of the presented model the reader is referred to [9]. It is important to note that the spatial regularization is here only introduced for the purpose of the mathematical analysis.In practice this regularization step is ignored (or, equivalently, σ is set to 0), which allows to obtain non trivial steady states.

Chapter 3

The application of the diffusion-delay anisotropic equation model for color image filtering

Denoising methods for color images can essentially be classified into four categories: neighborhood filters, frequency domain methods, variational PDE based methods and non-local methods. The most commonly used neighborhood filters include blurring filters, median filter or variations of it, and others such as the filters described in [29, 30]. Neighborhood filters are easy to implement and fast, and in some applications they can be effective, although in general their effectiveness is limited. In frequency domain methods a wavelet, DCT or other type of transformations is first performed. A filter is then applied to the transformation. The most common frequency domain method is the wavelet thresholding. The wavelet thresholding method assumes that noise appear in the wavelet transformation as small nonzero coefficients in the high frequency range and sets them to zero. The wavelet thresholding method is very effective in removing noise, and very fast. It is still perhaps the best "quick and easy" denoising scheme. However, it suffers from Gibbs oscillations at discontinuities. These oscillations can be reduced, although not eliminated, by using soft wavelet thresholing [11, 12] and translational invariant wavelet thresholding [7]. Methods that are not PDE based and do not fit the description of the other two, including some statistical methods, can be classified as non-local denoising methods, see e.g. [1, 40, 13, 22, 32]. These methods are typically

slower, and some of them assume certain statistical properties are known. Given the right images, non-local methods can yield excellent results. For example, the non-local method developed in [1] and its refinement in [32] work very well for images with repeat patterns or large homogeneous areas. Variational PDE based methods are quite effective and easy to implement but their performance depends on the noise level in the case of images corrupted with high level of noise it retains some noise or oversmooth the image.

The proposed method for color image denoising is based on the time-delay anisotropic diffusion model described in Chapter 2 i.e. the extension of the model (2.10)-(2.11). In the RGB color space we denoise each of the three channels to complete the filtering of the color image.

3.1 Application

The proposed method for color image denoising is based on the time-delay anisotropic diffusion model described in Chapter 2 i.e. it is the extension of the model. In the RGB color space we denoise each of the three channels to complete the denoising of the color image. So to clarify the application of the discussed model for color images we need to discuss the choise of various parameters in the model (2.10)-(2.11). As it was mentioned above, the parameter σ is taken equal to zero. Firstly, it is important to explain the choice of the F in the right hand side of the time-delay regularization equation (2.11).

The goal of the choices is preventing diffusion across the significant edges of the image. The distinction between significant edges and high gradient zones resulting from noise is based on averaging. So, the first choice is made as orthogonal projection on the direction orthogonal to the gradient of the image:

$$F_0(\bigtriangledown u) = P_{\bigtriangledown u^{\perp}} \tag{3.1}$$

In two dimensions, this means that, with the notation $\bigtriangledown u = (u_1, u_2)$

$$F = |\bigtriangledown u|^{-2} \begin{bmatrix} u_2^2 & -u_1 u_2 \\ -u_1 u_2 & u_1^2 \end{bmatrix} \tag{3.2}$$

Also it is necessary to introduce a time scale factor τ in the relaxation equation (2.11) so that the evaluation equation for the diffusion tensor reads

$$\frac{\partial L}{\partial t} + \frac{1}{\tau} L = \frac{1}{\tau} F_0(\bigtriangledown u) \tag{3.3}$$

Equation (3.3) means that the diffusion direction is a time average, over a time scale of τ, of the directions perpendicular to the edges, while the effect of the initial choice for L fades away. In comparison with reaction-diffusion equation where mentioned above directions were based on space averages of the gradients, a pure diffusion equation where the same directions based on time averages of the gradients is more tractable.

As the first choice of F leads to the many steady states (any image together with the diffusion tensor along directions perpendicular to the edges is a steady state) which in its term leads to the fast convergence of the filter. As a result, the processed image retains a significant amount of noise. It is notably in the case of images corrupted with high noise level.

To overcome this difficulty Cottet and El Ayyadi [9] proposed to select the steady states on the basis of a contrast threshold parameter. They define

$$F_s(\bigtriangledown u) = \begin{cases} P_{\bigtriangledown u^\perp} \quad if \quad |\bigtriangledown u| \geq s \\ \frac{3}{2}(1 - \frac{|\bigtriangledown u|^2}{s^2})Id + \frac{|\bigtriangledown u|^2}{s^2} P_{\bigtriangledown u^\perp} \quad if \quad not \end{cases} \tag{3.4}$$

When the gradients are not large enough the diffusion matrix will thus still be fed with isotropic diffusion, allowing to further filter the image away from the edges.

By considering image such that $L \bigtriangledown u = 0$ where L has the form of the the right hand side above, one easily finds that the steady states resulting from this model are images made of homogeneous patterns separated by fronts of stiffness larger than s.

The model will produce such patterns on its asymptotic states and the relaxation parameter τ determines the scales of these patterns. The smaller the parameter τ is chosen, the smaller the features of the pictures can be preserved. However this relaxation parameter should be large enough to allow the initial isotropic diffusion to eliminate noise.

3.2 Description of the numerical procedure

Equation (2.10) is discretized by using classical one-sided difference schemes for the operators divergence and $\bigtriangledown$, together with an explicit time-discretization. If u_{pq} is the value of the u at the pixel $(x = ph, y = qh)$ we set

$$\triangle^x_+ u_{p,q} = u_{p+1,q} - u_{p,q}, \quad \triangle^x_- u_{p,q} = u_{p,q} - u_{p-1,q} \tag{3.5}$$

and similar formulas for finite-differences in the y direction. Let us denote by

$$[L_{pq}]_{xx}, \quad [L_{pq}]_{yx}, \quad [L_{pq}]_{xy} \quad [L_{pq}]_{yy} \tag{3.6}$$

the four entries of the tensor L at the pixel (ph, qh). Then (2.10) is solved by

$$\frac{u_{pq}^{n+1} - u_{pq}^{n}}{\triangle t} - \frac{\triangle_{+}^{x}([L_{pq}]_{xx}^{n}\triangle_{-}^{x}u_{pq}^{n} + [L_{pq}]_{xy}^{n}\triangle_{-}^{y}u_{pq}^{n})}{h^2} - \frac{\triangle_{+}^{y}([L_{pq}]_{yx}^{n}\triangle_{-}^{x}u_{pq}^{n} + [L_{pq}]_{yy}^{n}\triangle_{-}^{y}u_{pq}^{n})}{h^2} = 0, where the time step is denoted by \triangle t. \tag{3.7}$$

As for (3.3), it is solved by the implicit scheme:

$$\frac{L^{n+1} - L^{n}}{\triangle t} + \frac{L^{n+1}}{\tau} = \frac{F^{n}}{\tau} \tag{3.8}$$

which gives

$$L_{pq}^{n+1} = (\frac{1}{1+\beta})(\beta L_{pq}^{n} + F_{pq}^{n}) \tag{3.9}$$

where $\beta = \frac{\tau}{\triangle t}$ and F_{pq} is computed on the basis of (3.4) with gradient obtained through centered finite difference. Note that using an implicit scheme for (3.3) (at no additional cost compared to an explicit scheme) allows to unconditionally preserve the positivity of the tensor L.

Color images can be described as a tree-band monochrome image data, where each band corresponds to a different color. So typical color images are represented as red, green, and blue or RGB images. Using the 8-bit monochrome standard as a model, the corresponding color image would have 24 bits/pixel – 8-bit for each of the three-color bands (red, green, and blue). In the purpose of applying the numerical procedure described above it is necessary to split the observed image on its tree channels (Red, Green and Blue). After that we will be able to filter each channel separately and combine denoised channels to complete the filtering process for color images.

We should note that all model parameters that are described in previous section can be set for each channel with different values as the noise level and contrast values of the channels can differ.

The algorithm below summarizes the different steps of the method.

Algorithm 1. Proposed algorithm for color image denoising

Step 1: Initialization

Input: A noisy image u_0 as a $N \times M \times 3$ matrix, diffusion matrix L_0 contrast parameter $s_i > 0$, $\tau_i > 0$, time-step parameters $\delta_t > 0$ and algorithm stopping criterion, $b_i > 0$ $i = 1, 2, 3$

Step 2: splitting of the observed image to its tree channels (Red, Green and

Blue)
Step 3: While $|u(t) - u(t-1)| \geq b$ do
3.1. Computation of the gradient using differences (3.7)
3.2. Computation of the F using (3.4)
3.3. Reconstruction of the diffusion tensor using (3.9)
3.4. Image update
Step 4: Applying the iterative scheme to all three channels
Step 5: Reconstruction of the image from r,g,b channels
Output: The denoised image $u(T)$ as an $N \times M \times 3$ matrix.

The algorithms was implemented in the form of C++ script. Some numerical experiments on original images are provided in Chapter 4.

3.3 Error metrics

In order to evaluate the quality of the processed images we will use standard error metrics. The better a reconstructed image resembles the original one, the bigger should be the value produced by this metrics. A common measure used for this purpose is the peak signal to noise ratio (PSNR). In addition, it is also simple to calculate but it has only a limited, approximate relationship with the perceived errors noticed by the human visual system. This is why higher PSNR values imply closer resemblance between the reconstructed and the original images, but they do not provide a guarantee that viewers will like the reconstructed image.

Denoting the pixels of the original image by P_i and the pixels of the reconstructed image by Q_i we first define mean square error (MSE) between the two images as

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(P_i - Q_i)^2 \tag{3.10}$$

It is the average of the square of the errors (pixel differences) of the two images. The root mean square error (RMSE) is defined as a square root of the MSE and the PSNR is calculated as

$$PSNR = 20log_{10}\frac{max_i P_i}{RMSE} \tag{3.11}$$

The PSNR is expressed in decibels.

Chapter 4

Experimental results

In this chapter we first recall different common type of noise that one has to deal with in image processing and we discuss the behavior of the method on these types of noise. We also discuss the choice of the different parameters of the methodsand of the stopping criterion and how they affect the results of the algorithm. Finally, we illustrate our method by compring its results with Chambolle's projection algorithm for total variation denoising and the DA3D algorithm.

4.1 Experiments with different types of noise

Noise is considered as an undesirable effect in image processing which can damage the image at the time of its capturing or further transmission. There are numerous sources and reasons of image noise which can be classified as Impulse noise (Salt and pepper noise), Amplifier noise (Gaussian noise), Shot noise, Quantization noise (uniform noise), Multiplicative noise (Speckle noise) and Periodic noise [34]. The goal of image processing is often to remove a specific kind of noise.

4.1.1 Impulse Noise (Salt and Pepper Noise)

The impulse noise has various denominations in the literature, like spike noise, random noise or independent noise. It is also well-known as a salt and pepper noise as black and white dots appear [4] on the image as a result of noise. This noise increases in the image as a result of sharp and sudden changes of image signal.

The first experiment illustrates the behavior of the algorithm when the color image is corrupted with Salt and Pepper noise of deviation $\sigma = 30$ (Figure 4.1).

(a) Original image
.
.

(b) Noisy image
RMSE=63.865
PSNR=11,99

(c) Denoised image
RMSE=9.09
PSNR=28.94

Figure 4.1: From left to right, original image, noisy image corrupted with Salt and Pepper noise with parameter $\sigma = 30$ and filtered image by proposed algorithm.

The root mean-square error (RMSE) and Peak signal-to-noise ratio (PSNR) are two error metrics used to compare image processing quality. Denoised image shows that the PSNR is significantly improved while the RMSE decreases more than six times. The RMSE represents the cumulative squared error between the denoised and the original image, whereas PSNR represents a measure of the peak error. The lower the value of RMSE, the lower the error. The higher the PSNR, the better the quality of the denoised image. Although Some blurring effect is visible in the denoised image, it appears that the proposed algorithm provides good results both visually and in terms of PSNR and RMSE. We can conclude that the proposed algorithm performs well as a tool for removing the noise of salt and pepper type.

4.1.2 Gaussian Noise (Amplifier Noise)

This noise model is additive in nature [36] and follows a Gaussian distribution. This means that each pixel in the noisy image is the sum of the true pixel value and a random, Gaussian distributed noise value.

The second experiment demonstrates the behavior of the proposed algorithm in the case of Gaussian noise. As an example, there was taken an image which was captured with Gaussian noise (noise level $\sigma = 30$) and then processed through our algorithm (Figure 4.2).

The results that were obtained after our second experiment with Gaussian noise show that the denoised image provides significant performance on the visual level. The PSNR is also very high which serves as an evidence of the best quality of the processed image. Furthermore, RMSE falls

(a) Original image
.
.

(b) Noisy image
RMSE=72.68
PSNR=10,96

(c) Denoised image
RMSE=8.01
PSNR=63.08

Figure 4.2: From left to right, original image, noisy image corrupted with Gaussian noise with parameter $\sigma = 30$ and filtered image by the proposed algorithm.

down dramatically. This confirms that our algorithm behaves much better in removing Gaussian noise in comparison with Salt and Pepper noise.

4.1.3 Poisson Noise (Photon Noise)

Poisson or shot photon noise is the noise that can result when the number of photons sensed by the sensor is not sufficient to provide detectable statistical information [36]. This noise has a root mean square value proportional to square root intensity of the image. Different pixels are corrupted by independent noise values [4].

The third experiment with Poisson noise (Figure 4.3) confirms that the proposed algorithm works well to remove this type of noise.

4.1.4 Stopping criterion for the algorithm

The question of finding an optimal stopping criterion has been studied in several previous works. Capuzzo, Dolcetta and Ferretti [20] determine the optimal time by finding the minimum of a performance index which balances the computing and stopping costs. It is then applied to the regularized Perona-Malik equation. Their method requires a constant that is found by experimentation using a typical image with similar details and discontinuities as the image to be processed. This is a rather vague requirement and they demonstrate that one only needs some approximation to the constant. Sporring and Weickert [23]choose the stopping criteria based on

(a) Original image
.
.

(b) Noisy image
RMSE=20.54
PSNR=16.77

(c) Denoised image
RMSE=11.08
PSNR=26.87

Figure 4.3: From left to right, original image, noisy image corrupted with Poisson noise and filtered image by proposed algorithm.

the signal to noise ratio and the relative variance at time t and the initial image. Mrazek and Navara [39] extend this idea and choose the stopping criteria so that the correlation of the signal $u(T)$ and noise $u(0) - u(T)$ in the filtered image is minimized. This is applied to several nonlinear filters both isotropic and anisotropic. Gilboa, Sochen and Zeevi [17] also choose the optimal stopping criteria to obtain a minimal SNR, i.e. one stops the process when filtering more signal than noise. This is done by estimating the covariance of the image and the noise.

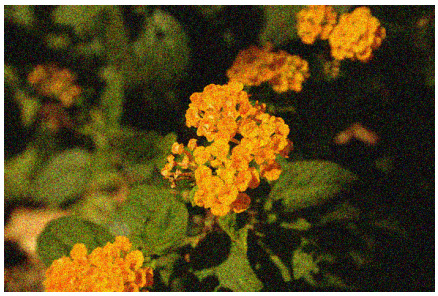
As already said, one important feature of our proposed algorithm is that it exhibits nontrivial steady states. This allows to choose a the stopping criterion on a very simple basis, by looking at residual difference between the processed images at successive iterations. We stop the algorithm as soon as this value goes below a threshold b. For example, we may set $b = 10^{-4}$.

Color images can be described as a tree-band monochrome image data, where each band corresponds to a different color. So typical color images are represented as red, green, and blue or RGB images. Using the 8-bit monochrome standard as a model, the corresponding color image would have 24 bits/pixel – 8-bit for each of the three-color bands (red, green, and blue). In the purpose of applying the stopping criterion method described above it is necessary to split the observed image to its tree channels (Red, Green and Blue). Also during the experiments, it was noticed that the noise level of three colors are different, the blue channel is the most noisy channel among others. It thus can be advisable to set different $b_i, \quad i = 1, 2, 3$

thresholds for each channel as a stopping criterion.

The next experiment shows the influence of the value of residual threshold on the result for the flower image, using the scheme that was described above. In this experiment the same threshold is chosen for all channels (Figure 4.4). One can see that taking a very small threshold value can result in undesired blurring of the image.

(a) Original image
.



(b) Noisy image
RMSE=32.92 PSNR=17.78

(c) Denoised image with b=0.0001
RMSE= 1.30 PSNR=45.8512

(d) Denoised image with b=0.00001
RMSE=7.22 PSNR=30.97

Figure 4.4: Top left: original image, top-right : noisy image with Gaussian noise with standard deviation = 40, bottom-left : processed image and bottom-right : processed image with $b = 0.0001$ when the PSNR has its maximum value and bottom-right : processed image with $b = 0.00001$.

4.1.5 Discussion of the model parameters

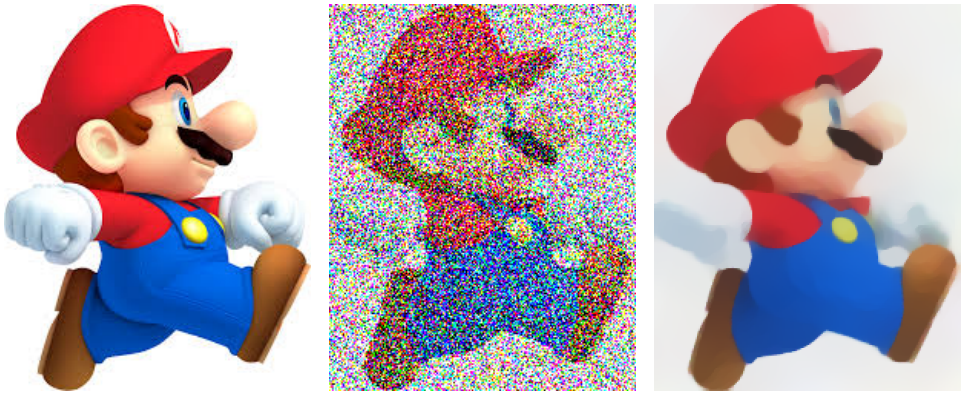
As it was explained above in Section 3, steady states are selected on the basis of a contrast parameter s. If $s = |\bigtriangledown u|$, then $L = F_s(\bigtriangledown u)$ and the algorithm stops. Conversely, if the contrast threshold parameter s is larger than $\bigtriangledown u$ and gradients are not large enough the diffusion matrix will still

be fed with isotropic diffusion which allow to continue the filtering process.

The relaxation parameter τ determines the amount of Gaussian diffusion which will remain in the filter before the image settles on a steady state. In practice this means that it must be chosen according to the noise level.

So far in in all our experiments we have taken a fixed value $\tau = 10$ and $s = 5$. In the next experiment we deal with a very noisy image. Due to the high level of noise we have set a higher values for our parameters (Figure 4.5). In this case the original image is corrupted with a Gaussian noise with standard deviation $\sigma = 60$ and the filter parameters were $s = 7$ and $\tau = 20$.

It is evident from the Figure 4.5. that the denoised image is somewhat blurred but the significant edges are reasonably preserved.



(a) Original image .

(b) Noisy image, $\sigma = 60$.

(c) Denoised image $s = 7$, $\tau = 20$

Figure 4.5: From left to right, original image, noisy image corrupted with Gaussian noise with parameter $\sigma = 60$ and filtered image by proposed algorithm.

In the next experiments we compare the results of the algorithm for different values of parameter τ to show its influence on the model. All experiments were led on the Flower image from Figure 4.7. corrupted with Gaussian noise of standard deviation $\sigma = 50$. In the first case we set equal values $\tau_{rgb} = 20$ for all three channels, in the second case we set values of relaxation parameter as follows $\tau_r = 10$, $\tau_g = 10$, $\tau_b = 20$, in the third case we used the reinitialization method suggested in [9] i.e. once the algorithm reaches the prescribed residual, say at time T, it is restarted with the initial image u_0 and the diffusion tensor obtained at the end of the previous cycle

L_T (Figure 4.6).

(a) Proposed with equal τ PSNR=18.52

(b) Proposed with different τ PSNR=20.34

(c) Processed image PSNR=20.92

Figure 4.6: From left to right a) with equal values of parameter $\tau_{rgb} = 20$ for all channels, b)with different values of parameter τ_{rgb}, $\tau_r = 10$ $\tau_g = 10$ $\tau_b = 20$, c) with the same values of parameter τ and using reinitialization technique

According to the last experiment It is evident that in the case of high level of noise it is advisable to choose different values of τ for each channel (Figure 4.6 (b)) or to apply a reinitialization technique (Figure 4.6.(c)).

4.2 Comparison with other Denoising Algorithms

Denoising of an image is an essential part of image Reconstruction process. Many algorithms exist to solve this problem e.g. neighborhood filters ([29, 30]), the Wiener local empirical filter [29], the discrete universal denoiser [14], Chambolle algorithm [21] and so on. In this section we compare our algorithm, both in terms of quality metrics and computational complexity, with Chambolle and D3AD algorithms.

4.2.1 Chambolle projection algorithm

Chambolle's projection algorithm for image denoising is based on Rudin-Osher-Fatemi (ROF) model, where the denoised image u is the solution of the following minimization problem:

$$min_{u \in BV(\Omega)} \int_{\Omega} | \bigtriangledown u(x)| dx \tag{4.1}$$

subject to the next constraints

$$\int_{\Omega} u(x)dx = \int_{\Omega} f(x)dx \quad and \quad \int_{\Omega} |u(x) - f(x)|^2 dx = \sigma^2 |\Omega| \tag{4.2}$$

The first constraint assumes that the noise has zero mean while the second constraint uses a priori information on the noise standard deviation σ. Later, it was proved that (4.1) and (4.2) are linked to the next unconstrained minimization problem [2]:

$$min_{u \in BV(\Omega)} \int_{\Omega} | \bigtriangledown u(x)| dx + \frac{\lambda}{2} ||u - f||_2^2 \tag{4.3}$$

for an appropriate Lagrange multiplier $\lambda > 0$. With the following notations:
f-observed noisy image
$BV(\Omega)$ - the space of functions of bounded variations and defined as

$$BV(\Omega) := u \in L^1(\Omega) : \int_{\Omega} | \bigtriangledown u| < \infty \tag{4.4}$$

The first term in (4.3) is a smoothing term and the second one measures the fidelity to the data. The parameter λ controls the trade off between the regularity and fidelity terms. As λ gets smaller the weight of the regularity term increases. Therefore λ is related to the degree of filtering of the solution of the minimization problem.

Chambolle proposed the semi-implicit gradient descent scheme described in Algorithm 1 for computing the discrete minimizer of (4.3).

Algorithm 1. Chambolle's projection algorithm for gray-scale TV denoising

---

Input: A noisy image $f(i,j)$ as a NxN matrix, $\lambda > 0$, time-step parameters $\delta_t > 0$ and algorithm tolerance $tol > 0$
Output: The denoised image $u(i,j)$ as an NxN matrix.
while $max_{1 \leq i,j \leq N} |p_{(i,j)}^{n+1} - p_{(i,j)}^{n}| > tol$ do
for all pixel (i,j) in the image do
$p(i,j) \leftarrow \frac{p(i,j) + \delta_t D(divp - \lambda f)(i,j)}{1 + \partial_t |D(divp - \lambda f)(i,j)|}$
end for
end while

return $u = f - \frac{1}{\lambda} divp$

The Vectorial Rudin-Osher-Fatemi model (VROF) for color image denoising is

$$min_{u \in BV(\Omega), R^M} \int_\Omega | \bigtriangledown u| dx + \frac{\lambda}{2} ||u - f||^2_{L^2(\Omega, R^M)} \tag{4.5}$$

where $\int_\Omega | \bigtriangledown u|$ is an alternative notation for VTV, $f = (f_1, ..., f_M)$ is the given noisy vector-valued image and the $L^2(\Omega, R^M)^2$ norm is defined as

$$||u - f||^2_{L^2(\Omega, R^M)} = \int_\Omega \sum_{m=1}^{M} |u_m - f_m|^2 dx \tag{4.6}$$

Chambolle's projection algorithm in the vectorial case can be obtained so that, for any color image $f \in L^2(\Omega, R^M)$, the minimizer of (4.5) is computed as

$$u = f - \pi_{\frac{1}{\lambda} K_{VTV}(f)} \tag{4.7}$$

where $K_{VTV} = v \in L^2(\Omega, R^M) :< v, u >_{L^2(\Omega, R^M)} \leq VTV(u) \forall u \in L^2(\Omega, R^M)$. And **Algorithm 1** is adapted for calculating $\pi_{\frac{1}{\lambda} K_{VTV}(f)}$ in the discrete setting:

$$p_m^{n+1}(i,j) = \frac{p_m^{n+1}(i,j) + \delta_t D(divp_m^n - \lambda f_m)(i,j)}{1 + \partial_t \sqrt{\sum_{m=1}^{M} |D(divp_m^n - \lambda f_m)(i,j)|^2}} \quad \forall 1 \leq m \leq M \tag{4.8}$$

As in the case of anisotropic differential equations, straight edges are maintained because of their small curvature. However, details and texture can be oversmoothed if λ is too small. Nevertheless, there is no definite formula to find λ corresponding to a particular value of σ. So Chambolle proposed [6] an algorithm to generate a sequence of values that converges to a unique λ such that $||u - f||_2^2 = \sigma^2$.

4.2.2 Comparison with Chambolle's algorithm

In order to evaluate our proposed algorithm, a comparison with Chambolles's projection algorithm is done. The source code of Chambolle's projection algorithm was taken from IPOL web-page [21]. Therefore, for comparison, there was used source code provided there and values of all parameters were taken as in the corresponding article. There were taken original images of different sizes and in advance we create from them noisy

images by adding Gaussian noise of several standard deviations. All experimental images with noise were processed by both algorithms. Figure 4.7 illustrates all images that were used in our experiments for comparison of two algorithms.

In addition, Table 4.1 illustrates RMSE and PSNR of noisy images as well as corresponding denoised images after applying both algorithms. It is evident from the table that our proposed algorithm gives better results than Chambolle's algorithm as the value of PSNR is lower and RMSE is higher in the case of Chambolle's algorithm. Furthermore, Chambolle's projection algorithm shows the worst results with image "Flowers", it can be explained with a fact that total variation cannot distinguish between huge gradients because of the noise singularities on the image such as vegetation texture [21].

(a) Nature 1920x1080

(b) Pinguin 300x300

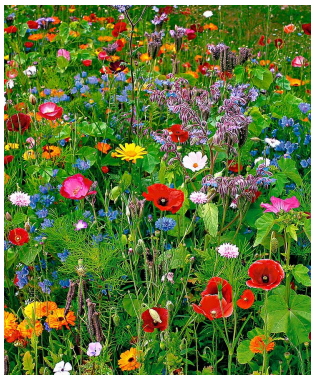
(c) Autoumn 2968x2190

(d) Dog 962x662

(e) Lena 512x512

(f) Dice 2160x1440

(g) Scientist 736x841



(h) Flower 736x888

(i) Mario 1473x1854

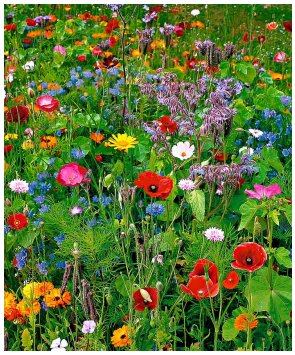
Figure 4.7: Original images used for the comparison between Chambolle's projection algorithm and the proposed algorithm.

Moreover, it is very important to compare visually the performance of both algorithms to see the characteristic artifacts of each one. For this purpose we have taken an image corrupted with Gaussian noise of deviation $\sigma = 50$ and denoised by both algorithms (Figure 4.8). It is obvious from the

Figure 4.8. that the Chambolle's algorithm does not work well in the case of high level of noise and that processed image still retains a substantial amount of noise, while our proposed algorithm performs better although the resulting image is a little bit blurred.

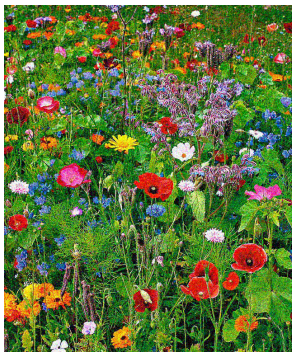
Images	Noisy image		Chambolle's algorithm		Proposed algorithm	
Metrics	RMSE	PSNR	RMSE	PSNR	RMSE	PSNR
Nature	18.63	22.73	8.79	29.25	2.86	30.98
Pinguin	27.48	19.35	8.01	30.06	4.04	34.99
Autoumn	28.54	19.02	16.82	23.61	6.62	26.12
Dog	25.95	19.85	6.20	32.28	1.57	44.19
Lena	28.99	18.88	10.50	27.70	3.85	30.41
Dice	25.63	19.95	8.79	29.25	3.59	37.03
Scientist	23.03	20.88	7.28	30.88	4.46	35.14
Flower	27.20	19.44	21.99	21.29	10.44	28.80
Mario	24.86	20.22	8.94	29.10	5.55	31.24

Table 4.1: Comparison between two algorithms with standard deviation $\sigma = 30$ are displayed.

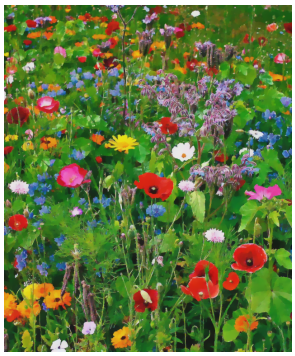


(a) Original image
.

(b) Noisy image
RMSE=43.31 PSNR=15.40



(c) Chambolle
RMSE=35.23 PSNR=17.19



(d) Proposed algorithm
RMSE=30.240 PSNR=18.52

Figure 4.8: Top-left : original image, top-right : noisy image with Gaussian noise with standard deviation $\sigma = 50$, bottom-left : processed image by Chambolle's and bottom-right : processed image by the proposed algorithm.

4.2.3 Comparison with Data Adaptive Dual Domain Denoising

DA3D (Data Adaptive Dual Domain Denoising) is one of the latest and most effective image denoising methods. It can also be called a "last step denoising" method that takes as input a noisy image and as a guide the result of any state-of-the-art denoising algorithm. DA3D is an iterative algorithm that uses a guide image (from a previous iteration) to determine spatially uniform regions to which Fourier shrinkage could be applied without introducing ringing artifacts [38, 27]. It is a "last step" denoising method that performs frequency domain shrinkage on shape-adaptive and data-

adaptive patches. DA3D consistently improves the results of state-of-the-art methods and many experiments involving different methods were taken as a guide for DA3D in the paper [27]. According to the results obtained in this paper, Non-Local Bayes (NL-Bayes) image denoising method showed the best results both visually and in terms of PSNR.

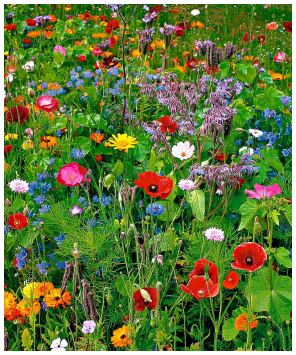
Non-Local Bayes (NL-Bayes) image denoising algorithm is an improved variant of the NL-means image denoising method. In this method, each patch is replaced by a weighted mean of the most similar patches presenting a neighborhood. The NL-Bayes strategy improves on NL-means by evaluating for each group of similar patches a Gaussian vector model. To each patch is therefore associated a mean (which would be the result of NL-means), but also a covariance matrix estimating the variability of the patch group. This permits to compute an optimal(in the sense of Bayesian minimal mean square error) estimate of each noisy patch in the group,by a simple matrix inversion [5]. NL-Bayes is very similar to many state of the art methods like (TSID, BM3D, BM3D-SAPCA). According to the experimental results provided in [5], Nl-Bayes is the best state of the art method for color image denoising in terms of PSNR and CPU time.

As DA3D is one of the latest and most effective image denoising method in combination with NL-Bayes, we made some experimental tests to evaluate our algorithm. A reliable implementation of DA3D can be found in the IPOL platform. We used the source code provided there and took noise-free images of different size. Both algorithms have been processed on the noisy images obtained from original ones after adding additive Gaussian noise of standard deviation $\delta = 50$.

The first experiment is made on the Flower image corrupted with Gaussian noise of standard deviation $\sigma = 50$. This image was chosen because it includes many small details and various gradients of color which makes it very challenging. The experiment is illustrated in Figure 4.9.

All three algorithms, namely Chambolle, proposed and DA3D algorithms were processed on Flower image corrupted with Gaussian noise of standard deviation $\sigma = 50$. Our proposed algorithm is processed applying once the reinitialization technique, which as we have seen gives more robust results with respect to the relaxation parameter. Visually it is evident that the proposed and DA3D algorithms perform well and look very similar to original one. In term of the PSNR metrics, DA3D performs slightly better.

Our last experiment is based on the set of the images displayed in Figure 4.7. All three algorithms were ran on the noisy image created from original images by adding Gaussian noise of standard deviation $\sigma = 30$.



(a) Original image

.

(b) Noisy image PSNR=15.40

(c) Chambolle PSNR=17.19

(d) Proposed PSNR=20.92

(e) DA3D PSNR=22.40

Figure 4.9: From left to right in the top row a) original image, b) noisy image captured by Gaussian noise with standard deviation $\sigma = 50$, and in the bottom row c) denoised image by Chambolle's d) denoised images by proposed algorithm and e) denoised image by DA3D

It is evident from the Table 4.2. that DA3D provides the best results in terms of the quality metrics PSNR, but is the most time consuming. Its computational complexity results from the fact that keeping track of the minimum weight can require scanning of the whole image. In contrast, Chambolle's algorithm and our proposed algorithm are quite fast.

Images	DA3D		Chambolle's algorithm		Proposed algorithm	
Metrics	CPU	PSNR	CPU	PSNR	CPU	PSNR
Nature	3.25	31.02	0.62	29.25	1.25	30.98
Pinguin	0.46	34.50	0.25	30.06	0.11	34.99
Autoumn	0.49	26.19	0.38	23.61	0.16	26.12
Lena	0.34	30.52	0.24	27.70	0.17	30.41
Mario	0.56	31.74	0.25	29.10	0.19	31.24

Table 4.2: Comparison in various experiments of CPU time corresponding to the three algorithms with standard deviation $\sigma = 30$ are displayed.

Chapter 5

Conclusion

We have studied the capabilities of a class of time-delay anisotropic nonlinear diffusion PDEs for color image filtering. The diffusion equations are solved independently on each channel of the image.

Despite of the simplicity of the proposed approach, it gives excellent results in color image denoising, visually and in terms of quality metrics.

In a first set of experiments we have investigated the performance of the algorithms for several types of noise. We have shown that the best performance was obtained in the case of Gaussian noise and the worse in the case of Poisson noise. These experiments allowed also to discuss the choice of the two parameters that define the algorithm, the contrast coefficient which selects the steady states and the relaxation time which controls the amount of smoothing.

We have further evaluated our algorithm against Chambolle's projection algorithm and DA3D (Data Adaptive Dual Domain Denoising) algorithm, using source codes provided by IPOL.

These experiments show that, even in presence of a large amount of noise, our algorithm yields a rather good trade-off between quality, evaluated both visually and in terms of the quality metrics, and computational complexity.

Bibliography

[1] B. Coll A. Buades and J. M. Morel. A review of image denoising algorithms, with a new one, multiscale model. simul. *electronic*, 4(2):490–530, 2005. 15, 16

[2] P.-L. Lions A. Chambolle. Image recovery via total variation minimization and related problems. *Numeric Mathematic*, 76:167–188, 1997. 27

[3] B. Benhamouda. Parameter adaptation for nonlinear diffusion in image processing. *Master's thesis, University of Kaiserslautern, Kaiserslautern*, 1994. 10

[4] Charles Boncelet. Image noise models. *in Alan C.Bovik. Handbook of Image and Video Processing*, 2005. 20, 22

[5] A. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8:769–698, 1986. 8, 33

[6] A. Chambolle. An algorithm for total variation minimization and application of mathematical imaging and vision. *Journal of Mathematical Imaging and Vision*, 20:89–97, 2004. 1, 28

[7] R. Coifman and D. Donoho. Translation-invariant de-noising, in wavelets and statistics. *Springer Verlag*, pages 125–150, 1995. 15

[8] G.-H. Cottet. Neural networks: Continuous approach and applications to image pro- cessing. *J. Biological Systems*, (3), 1995. 7, 11

[9] G. H. Cottet and M. El Ayyadi. A voltera type model for image processing. *IEEE Transactions on Image Processing*, 7:292–303, 3 1998. 7, 11, 13, 14, 17, 25

[10] G.-H. Cottet and L.Germain. Combining nonlinear diffusion with reaction for image processing. *Math.Comput*, 61:659–673, 1993. 7, 11, 12

[11] D. L. Donoho. De-noising by soft-thresholding. *IEEE Trans. Information Theory*, (41(3)):613–627, 1995. 15

[12] D. L. Donoho and I. M. Johnstone. Ideal spacial adaption by wavelet shrinkage. *Biometrika*, (81):425–455, 1994. 15

[13] E.Ordentlich, S.Verdu G.Seroussi, M.Weinberger, and T.Weissman. A discrete universal denoiser and its application to binary images. *Proc. IEEE ICIP*, 1:117–120, 2003. 15

[14] E.Ordentlich, S.Verdu G.Seroussi, M.Weinberger, and T.Weissman. A discrete universal denoiser and its application to binary images. *in Proc. IEEE ICIP*, 1:117–120, 2003. 26

[15] J.-M. Morel F. Catté, P.-L. Lions and T. Coll. Image selective smoothing and edge detection by nonlinear diffusion. *SIAM Journal of Numerical Analysis*, 29(1):182–193, 1992. 6, 10

[16] J. Fröhlich and J. Weickert. Image processing using a wavelet algorithm for nonlinear diffusion. *Report 104, Laboratory of Technomathematics, University of Kaiserslautern, Kaiserslautern*, 1994. 10

[17] Zeevi G.Gilboa, N.Sochen. Estimation of optimal pde-based denoising in the snr sense. *IEEE Trans. Image Proc.*, (15):2269–2280, 2006. 23

[18] K. Höllig. Existence of infinitely many solutions for a forward-backward heat equation. *Transactions of the American Mathematical Society*, 278:299–319, 1983. 10

[19] K. Höllig and J.A. Nohel. A diffusion equation with a non-monotone constitutive function. *In J.M. Ball, editor, Proceedings of NATO/London Mathematical Society Conference on Systems of Partial Differential Equation*, page 409–422, 1983. 10

[20] Ferretti I.Capuzzo Dolcetta. Optimal stopping time formulation of adaptive image filtering. *Appl.Math*, (43), 2001. 22

[21] Catalina Shert Joan Duran, Bartomeu Coll. Chambolle's projection algorithm for total variation denoising. *Image processing on line (IPOL)*, 2013. 1, 7, 26, 28, 29

[22] M.Wainwright J.Portilla, V.Strela and E.P.Simoncelli. Image denoising using scale mixtures of gaussians in the wavelet domain. *IEEE Transactions on Image Processing*, 12(11):1338–1351. 15

[23] J. Weickert J.Sporring. Information measures in scale-spaces. *IEEE Trans. Inf. Theory*, (45):1051–1058, 11 1999. 22

[24] J.Weickert. Scale-space properties of nonlinear diffusion filtering with a diffusion tensor. *prepint*, 1996. 7

[25] J.Weickert and C.Schnörr. Pde-based preprocessing of medical images. *IKunstliche Intelligenz*, 3:5–10, 2000. 10

[26] S. Kichenassamy. The perona-malik paradox. *SIAM Journal of Applied Mathematics*, (57(5)):1328–1342, 1997. 10

[27] J.J. Koenderink. The structure of images. biological cybernetics. *Journal of Mathematical Imaging and Vision*, 50:363–370, 1984. 8, 32, 33

[28] P.-L. Lions L. Alvarez and J.-M. Morel. Image selective smoothing and edge detection by nonlinear diffusion. *SIAM Journal of Numerical Analysis*, 29(3):845–866, 1992. 6

[29] L.P.Yaroslavsky. Digital picture processing - an introduction. *Springer Verlag*, 1985. 15, 26

[30] L.Yaroslavsky and M.Eden. Fundamentals of digital optics. *Birkhauser*, 1996. 15, 26

[31] D. Marr and E. Hildreth. Theory of edge detection. in proceedings of the royal society b. *Royal Society Publishing*, 207:187–217, 1980. 8

[32] M.Mahmoudi and G.Sapiro. Fast image and video denoising via nonlocal means of similar neighborhoods. *IEEE Signal Processing Letter*, 12:839–842, 2005. 15, 16

[33] M.Nitzberg and T.Shiota. Nonlinear image filtering with edge and corner enhancement. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 3:5–10, 2000. 10

[34] Dr. Jahid Ali Mr. Rohit Verma. A comparative study of various types of image noise and efficient noise removal techniques. *International journal of advanced research in Computer Science and Software engineering*, 3(ISSN: 2277 128X):187–217, 2013. 20

[35] T. Shiota P. Perona and J. Malik. Anisotropic diffusion. *In B.M. ter Haar Romeny, editor, Geometry Driven Diffusion in Computer Vision*, 1:72–92, 1994. 10

[36] Mr. Pawan Patidar and et al. Image denoising by various filters for different noise. *International journal of advanced research in computer science and software engineering*, (4):292–303, 11 2010. 21, 22

[37] P. Perona and J. Malik. Scale space and edge detection using anisotropic diffusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12:629–630, 1990. 6, 9, 10

[38] Nicola Pierazza and Gabriela Facciolo. Data adaptive dual domain denoising: a method to boost state of the art denoising algorithms. *Image processing on line (IPOL)*, 2017. 1, 7, 32

[39] M.Navara P.Mrazek. Selection of optimal stopping time for nonlinear diffusion filtering. *Int. J. Comput. Vision*, (52):189–203, 2003. 23

[40] L. Shen R. Chan, T. Chan and Z. Shen. Wavelet algorithms for high-resolution image reconstruction. *SIAM J. Sci. Comput.*, 24(4):1408–1432, 2003. 15

[41] J. Weickert. Anisotropic diffusion in image processing. *PhD thesis, Universität Kaiserslautern, Kaiserslautern, Germany*, 1996. 10, 11

[42] A.P. Witkin. Space-scale filtering. *Bundy, editor, Proceedings of the Eighth International Joint Conference on Artificial Intelligence*, page 1019–1022, 1983. 8, 9

[43] A.P. Witkin. Scale-space filtering: A new approach to multi-scale description. *In Proceedings of the IEEE International Conference on Acoustic, Speech Signal Processing*, 9:150–153, 1984. 9